A Note on the Complexity of the Asymmetric Traveling Salesman Problem

Weixiong Zhang

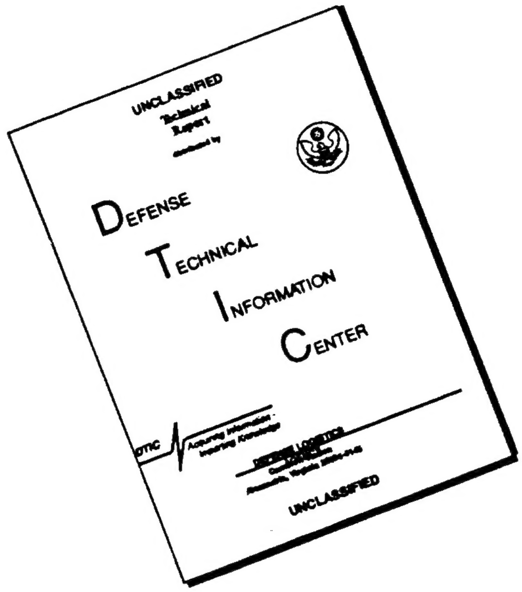
USC/Information Sciences Institute

May 1996

ISI/RS-96-445

This paper is to appear in the journal *Operations Research Letters,* published by North-Holland, Amsterdam.

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

REPORT DOCUMENTATION PAGE

FORM APPROVED
OMB NO. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 1996	Research Report

4. TITLE AND SUBTITLE

A Note on the Complexity of the Asymmetric Traveling Salesman Problem

5. FUNDING NUMBERS

DARPA:
MDA972-94-2-0010

6. AUTHOR(S)

Weixiong Zhang

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

USC INFORMATION SCIENCES INSTITUTE
4676 ADMIRALTY WAY
MARINA DEL REY, CA 90292-6695

8. PERFORMING ORGANIZATON REPORT NUMBER

ISI/RS-96-445

9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency
3701 N. Fairfax Drive
Arlington, VA 22203-1714

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

This paper is to appear in the journal *Operations Research Letters,* published by North-Holland, Amsterdam.

12A. DISTRIBUTION/AVAILABILITY STATEMENT

UNCLASSIFIED/UNLIMITED

12B. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

One of the most efficient approaches known for finding an optimal tour of the asymmetric Traveling Salesman Problem (ATSP) is branch-and-bound (BnB) subtour elimination. For two decades, expert opinion has been divided over whether the expected complexity of the ATSP under BnB subtour elimination is *polynomial* or *exponential* in the number of cities. We show that the argument for polynomial expected complexity does not hold.

14. SUBJECT TERMS

average-case complexity, combinatorial optimization, Traveling Salesman Problem

15. NUMBER OF PAGES

16

16. PRICE CODE

17. SECURITY CLASSIFICTION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reoprts. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to stay within the lines to meet optical scanning requirements.

Block 1. Agency Use Only (Leave blank).

Block 2. Report Date. Full publication date including day, month,a nd year, if available (e.g. 1 jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element numbers(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	- Contract	PR	- Project
G	- Grant	TA	- Task
PE	- Program Element	WU	- Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the repor.

Block 9. Sponsoring/Monitoring Agency Names(s) and Address(es). Self-explanatory

Block 10. Sponsoring/Monitoring Agency Report Number. (If known)

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of ...; To be published in... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD	- See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE	- See authorities.
NASA	- See Handbook NHB 2200.2.
NTIS	- Leave blank.

Block 12b. Distribution Code.

DOD	- Leave blank.
DOE	- Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
NASA	- Leave blank.
NTIS	- Leave blank.

Block 13. Abstract. Include a brief (Maximum 200 words) factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code (NTIS only).

Blocks 17.-19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contins classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

A Note on the Complexity of the Asymmetric Traveling Salesman Problem *

Weixiong Zhang †
Information Sciences Institute and
Department of Computer Science
University of Southern California
4676 Admiralty Way
Marina del Rey, CA 90292
Email: zhang@isi.edu

Abstract

One of the most efficient approaches known for finding an optimal tour of the asymmetric traveling salesman problem (ATSP) is branch-and-bound (BnB) subtour elimination. For two decades, expert opinion has been divided over whether the expected complexity of the ATSP under BnB subtour elimination is *polynomial* or *exponential* in the number of cities. We show that the argument for polynomial expected complexity does not hold.

*This paper is to appear in *Operations Research Letters.*
†The author was partially supported by DARPA Contract, #MDA972-94-2-0010.

1 Introduction

Given n cities $\{1, 2, \cdots, n\}$ and a cost matrix $(c_{i,j})$ which defines the cost between each pair of cities, the traveling salesman problem (TSP) [8] is to find a minimum-cost tour that visits each city once and returns to the starting city. When the cost $c_{i,j}$ from city i to city j is not necessarily equal to that from city j to city i, the problem is the asymmetric TSP (ATSP). It is well known that the ATSP is NP-hard [5].

One of the most efficient approaches for optimally solving the ATSP is branch-and-bound subtour elimination [2, 3] using the assignment problem as a lower-bound function. The solution of an assignment problem (AP) [11] is either a complete tour or a collection of disjoint subtours, which is a relaxation of the ATSP and can be solved in $O(n^3)$ time. Branch-and-bound (BnB) [2, 9] is one of the most general and efficient algorithms for finding the exact solutions of most combinatorial optimization problems. The algorithm first solves the AP for the n cities. If the solution is not a tour, then the problem is decomposed or expanded into subproblems by eliminating one of the subtours in the solution. A subproblem is then selected and the above process is repeated until each subproblem is solved, i.e., its AP solution is a tour, or until the costs of the AP solutions of the unsolved subproblems are greater than or equal to the cost of the best tour obtained so far.

Bellmore and Malone [3] argued that the ATSP can be solved in polynomial expected time using BnB subtour elimination. Their argument treated the process of BnB subtour elimination as a statistical experiment, with the i-th trial corresponding to the exploration of the i-th selected subproblem, and success corresponding to the i-th selected subproblem producing a tour of minimum cost. Let the sequence of subproblems selected by BnB subtour elimination be $X_0, X_1, X_2, X_3, \cdots$, and p_i be the probability that the AP solution of X_i is a tour, for $i = 0, 1, 2, \cdots$. They began with an observation that the probability p_0 of the AP solution to the original problem being a complete tour is approximately e/n, where n is the number of cities. They then assumed that the event that the AP of X_i is a tour is independent of whether the AP of X_j is a tour for $i \neq j$, and that $p_i \geq p_0$ for $i \geq 1$. Under these assumptions, the expected number of expanded subproblems is

$$\sum_{i=1}^{\infty} ip_i \prod_{j=0}^{i-1}(1 - p_j) \leq \sum_{i=1}^{\infty} ip_0(1 - p_0)^{i-1} = 1/p_0 = O(n)$$

for large n. Thus the expected running time is $O(n^4)$, as the APs can be solved in no more than $O(n^3)$ time [11]. Their experiments show an $O(n^{3.46})$ expected running time for $10 \leq n \leq 80$. Additional experiments reported by Smith et al. [14] show that the expected running time is $O(n^{3.2})$ for $30 \leq n \leq 200$. Smith [13] also argued, under many assumptions, that the expected complexity is $O(n^3 \ln(n))$.

Lenstra and Rinnooy Kan [10] pointed out that Bellmore and Malone's assumptions may not be valid. In particular, two trials may not be independent of each other, and p_i may not be greater than p_0, for $i \geq 1$. They also pointed out that to have an expected complexity of $O(n^c)$, for some constant $c > 0$, the following two conditions must hold. First, the AP solution of X_0, the original problem, must have a probability $p_0 = O(n^{-c})$ of being a tour. Second, the number of expanded subproblems whose probability of yielding a tour is less than p_0, must be a constant. In addition, Balas et al. [1, 2] carried out statistical analysis on the experimental data obtained from three efficient implementations of BnB subtour elimination. Their study concluded that over the range of $40 \leq n \leq 325$, the performance of BnB subtour elimination on randomly generated ATSPs can be described almost *equally well* by a polynomial function (αn^{β}), a superpolynomial function ($\alpha n^{\beta \log n}$), or an exponential function ($\alpha e^{\beta n}$).

In short, whether the expected complexity of the ATSP under BnB subtour elimination is a polynomial or an exponential function of the number of cities is an open question, on which the opinion of experts is divided [7].

In this paper, we disprove Bellmore and Malone's polynomial argument. We show that their critical assumption that $p_i \geq p_0$, for $i \geq 1$, does not hold under random ATSPs. In particular, we show that, even if the subproblems selected by BnB subtour elimination are assumed to be independent of each other, the algorithm expands more than $\ln(n)$ number of subproblems X_i for which $p_i < p_0$. This result supports and finalizes Lenstra and Rinnooy Kan's argument.

In Section 2, we briefly discuss the BnB subtour elimination algorithm and the analytic model we use. In Section 3, we disprove the polynomial argument. Our conclusion appears in Section 4.

2 Preliminaries

Let V be the set of n cities, and $C = (c_{i,j})$ be a cost matrix that specifies costs between all city pairs. Let Π_n be the set of all permutations of V, which defines all possible solutions to the assignment problem [11] on V. A permutation $\pi \in \Pi_n$ specifies an assignment of the cities, in which city $\pi(i)$ is assigned to city i, or $\pi(i)$ is the successor of city i. Thus, under π, a chain $1 - \pi(1) - \pi(\pi(1)) - \cdots \underbrace{\pi \cdots \pi}_{k}(1) - 1$ forms a subtour if $k < n$, or a complete tour if $k = n$. When $k = n$, π is called a cyclic permutation. Let $\Pi_n^* \subset \Pi_n$ be the set of all cyclic permutations of V. For $\pi \in \Pi$, define $f(\pi, C) = \sum_{i=1}^{n} c_{i,\pi(i)}$ to be the cost of π under cost matrix C. The ATSP is to find a *cyclic* permutation $\pi^* \in \Pi_n^*$, such that $f(\pi^*, C) = \min\{f(\pi, C) | \pi \in \Pi_n^*\}$. Similarly, the assignment problem (AP) seeks a permutation $\pi' \in \Pi_n$, such that $f(\pi', C) = \min\{f(\pi, C) | \pi \in \Pi_n\}$. In other words, the solution to the AP is a

collection of disjoint subtours. If the AP solution happens to be a complete tour, it is the solution to the ATSP as well.

BnB subtour elimination first solves the AP for all n cities, in time $O(n^3)$ [11]. If the solution is not a complete tour, then the problem is decomposed into subproblems by excluding some of the arcs in a subtour, which eliminates that subtour. Which subtour to choose, and how to eliminate a subtour, constitute the *branching rules*. One heuristic is to select a subtour with a minimum number of arcs that are not in the included arc set [4], so that the number of subproblems generated from a problem is minimized, as excluding one arc that is not included is sufficient for breaking a subtour. After a subtour is chosen, the subproblem should be decomposed such that no duplicate subproblems are generated, minimizing the total number of subproblems generated. One such branching rule was contributed by Carpaneto and Toth [4]. Their rule selects a subtour with the minimum number of arcs that are not in the included arc set, and decomposes the subproblem as follows. Let E be the excluded arc set, and I be the included arc set of the problem to be decomposed. Assume that there are t arcs of the selected subtour, $\{x_1, x_2, \cdots, x_t\}$, that are not in I. The rule decomposes the problem into t children, with the k-th one having excluded arc set E_k and included arc set I_k, such that

$$\left.\begin{aligned} E_k &= E \cup \{x_k\}, \\ I_k &= I \cup \{x_1, \cdots, x_{k-1}\}, \end{aligned}\right\} k = 1, 2, \cdots, t. \tag{1}$$

Since x_k is an excluded arc of the k-th subproblem generated, $x_k \in E_k$, and it is an included arc of the $k+1$-st subproblem, $x_k \in I_{k+1}$, a tour obtained from the k-th subproblem does not contain arc x_k, but a tour obtained from the $k+1$-st subproblem must have arc x_k. Thus a tour from the k-th subproblem cannot be generated from the $k+1$-st one, and vice versa. Therefore, the state space of the ATSP under BnB subtour elimination can be represented by a search tree without duplicates. This state-space tree can be explored by best-first search or depth-first search.

Following previous research [6, 7], in the following analysis we use a random cost matrix $(c_{i,j})$ whose elements are independently and uniformly chosen from the unit interval $[0, 1]$, and we do not enforce the symmetry $(c_{i,j} = c_{j,i})$ or the triangle inequality $(c_{i,j} + c_{j,k} \geq c_{i,k})$. Since we are interested in finding a cyclic permutation of the cities, we can set $c_{i,i} = \infty$ without loss of generality, for $i = 1, 2, \cdots, n$. The AP of a cost matrix with $c_{i,i} = \infty$ is called a *modified AP*, and a permutation with $c_{i,i} = \infty$ is called a *feasible permutation* [3].

Lemma 2.1 [3] *Given n cities, there are asymptotically $\lfloor n!/e + 0.5 \rfloor$ feasible permutations of the cities.* □

Lemma 2.2 [3, 7] *Given an $n \times n$ random matrix, the probability that a modified AP solution is a cyclic permutation is asymptotically e/n.* $\square$

Lemma 2.3 *Given an $n \times n$ random matrix, the expected number of subtours in a modified AP solution is asymptotically less than* $\ln(n)$*, and the expected number of arcs in a subtour of the solution is greater than* $n/\ln(n)$.

Proof: Solving the modified AP of a random cost matrix is equivalent to randomly selecting a feasible permutation. This is proved as follows. We first show that all feasible permutations are equally likely to have the minimum cost. Let Π_n be the set of all feasible permutations of n cities. Arbitrarily partition Π_n into two subsets $\Pi_n^{(1)}$ and $\Pi_n^{(2)}$, i.e., $\Pi_n = \Pi_n^{(1)} \cup \Pi_n^{(2)}$ and $\Pi_n^{(1)} \cap \Pi_n^{(2)} = \emptyset$. Then arbitrarily select one permutation π_1 from $\Pi_n^{(1)}$ and another one π_2 from $\Pi_n^{(2)}$. Without loss of generality, assume that π_1 and π_2 have k arcs (city pairs) *not* in common, and $n-k$ arcs in common, where $k = 1, 2, \cdots, n$. For a given cost matrix C, the cost $f(\pi_1)$ of π_1 and the cost $f(\pi_2)$ of π_2 can be written as

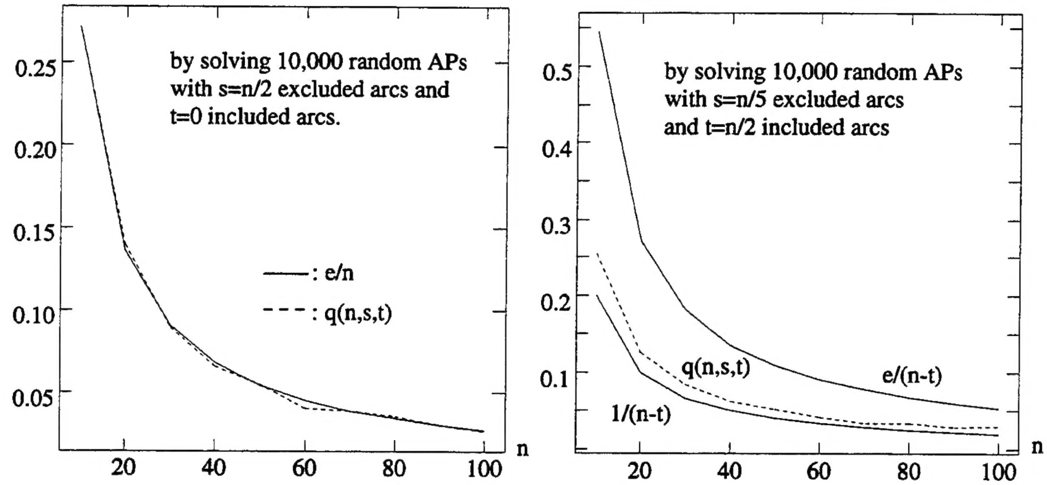
$$f(\pi_1) = \sum_{i=1}^{k} \varepsilon_i + \sum_{i=k+1}^{n} \varepsilon_i \quad \text{and} \quad f(\pi_2) = \sum_{i=1}^{k} \varepsilon_i' + \sum_{i=k+1}^{n} \varepsilon_i$$

where ε_i and ε_i' are different arc costs in π_1 and π_2. Obviously, whether $f(\pi_1) < f(\pi_2)$ or $f(\pi_1) > f(\pi_2)$ depends only on the different arc costs, $\sum_{i=1}^{k} \varepsilon_i$ and $\sum_{i=1}^{k} \varepsilon_i'$. Since ε_i and ε_i' are independent and identically distributed (*i.i.d.*) random variables, then the probability of $\sum_{i=1}^{k} \varepsilon_i > \sum_{i=1}^{k} \varepsilon_i'$ is the same as the probability of $\sum_{i=1}^{k} \varepsilon_i < \sum_{i=1}^{k} \varepsilon_i'$. In other words, each of the two permutations $\pi_1 \in \Pi_n^{(1)}$ and $\pi_2 \in \Pi_n^{(2)}$ has the same probability of being smaller. Consequently, with respect to the permutations in $\Pi_n^{(2)}$, all permutations in $\Pi_n^{(1)}$ are equally likely to have smaller costs. Because the subsets $\Pi_n^{(1)}$ and $\Pi_n^{(2)}$ were chosen arbitrarily, all feasible permutations are thus equally likely to have the minimum cost.

Furthermore, it is known that the expected number of subtours in a permutation asymptotically approaches $\ln(n)$ [12]. Thus the expected number of arcs in a subtour of an AP solution is $O(n/\ln(n))$. $\square$

3 On the Polynomial Argument

Let us assume the validity of Bellmore and Malone's assumption that two subproblems chosen by BnB subtour elimination are independent of each other. Then their polynomial argument critically depends on the assumption that a constant number of subproblems X_i are expanded whose probabilities p_i are less than p_0. In this section, we show that this argument does not hold even if the independence assumption is granted. We first prove the following lemma.



(a) q(n,s,t) when there is no included arc. **(b) q(n,s,t) when there are included arcs.**

Figure 1: Probability $q(n,s,t)$ that the AP solution is a tour

Lemma 3.1 *Given an $n \times n$ random cost matrix, let $q(n,s,t)$ be the probability that the solution of a modified AP, which has $s < n$ excluded arcs and t included arcs, is a tour. Then $q(n,s,t)$ is asymptotically*

$$e/n - o(1/n) < q(n,s,t) < e/n + o(1/n), \quad \textit{when } t = 0 \tag{2}$$

$$q(n,s,t) = \frac{\lambda}{n-t} + o(1/n), \quad \textit{when } t > 0 \tag{3}$$

where λ, $1 < \lambda < e$, is a constant.

Proof: See Appendix.

Since Lemma 3.1 is important to our analysis, we verified its correctness by experimentally solving 10,000 randomly generated APs for each $n \in \{10, 20, 30, \cdots, 100\}$. Our results are presented in Figure 1. Figure 1(a) shows that the experiments support (2) when $s = n/2$ excluded arcs and $t = 0$ included arcs are used. Figure 1(b) shows that the experiments support (3) when $s = n/5$ excluded arcs and $t = n/2$ included arcs are employed.

A subproblem X_i, which is selected by BnB subtour elimination and which generates an optimal tour, must be a minimum-cost leaf node in the search tree. Thus, the probability of X_i generating an optimal tour cannot be greater than the probability of X_i being a leaf node. Note that X_i can generate a complete tour and become a leaf node only when all its ancestors in the search tree do not generate complete tours but X_i does.

Lemma 3.2 *Suppose that two nodes chosen by BnB subtour elimination are independent of each other. Let p be the probability that a non-root node in the search*

tree is a leaf, and in particular let p_0 be the probability that the root is a leaf. For a non-root node with t included arcs, there exists a constant $0 < \delta < 1 - 1/e$ such that, if $t < \delta n$, then $p < p_0$, where n is the number of cities.

Proof: Consider a node Y of a search tree which is generated by equation (1). Let the number of included arcs of Y be t and the number of excluded arcs of Y be s, which is also the depth of the node in the search tree. Denote the nodes on the path from the root to the node Y as $Y_0, Y_1, Y_2, \cdots, Y_{s-1}, Y$, where Y_0 is the root. From equation (1), Y_i has i excluded arcs. In addition, let Y_i have t_i included arcs. By equation (1), $0 = t_0 \leq t_1 \leq t_2 \leq \cdots \leq t_{s-1} \leq t$. It is only when none of $Y_0, Y_1, \cdots, Y_{s-1}$ is a leaf node that Y exists in the tree. Therefore, the probability $q(n, s, t)$ of Y's AP solution yielding a complete tour is equal to the probability that all its ancestors do not generate complete tours and Y produces a complete tour. The probability that Y's parent Y_{s-1} does not yield a complete tour is $(1 - q(n, s-1, t_{s-1}))$, and that Y's grandparent Y_{s-2} does not is $(1 - q(n, s-2, t_{s-2}))$, and so on. Consequently, by the independence assumption, the probability p that Y exists and is a leaf is then

$$p = q(n,s,t) \prod_{i=0}^{s-1} (1 - q(n,i,t_i)).$$

By Lemma 3.1, we have

$$p = \left(\frac{\lambda}{n-t} + o(1/n)\right) \prod_{i=0}^{s-1} \left(1 - \frac{\lambda}{n-t_i}\right) = \frac{\lambda}{n-t}\left(1 - \sum_{i=0}^{s-1} \frac{\lambda_i}{n-t_i}\right) + o(1/n),$$

where $\lambda_0 > \lambda_1 > \cdots > \lambda_{s-1} > \lambda > 1$ are constants. It can be shown by induction that

$$\sum_{i=0}^{s-1} \frac{\lambda_i}{n-t_i} = \frac{\lambda' s}{n-t'}; \qquad \text{where} \quad \lambda' = \frac{1}{2}\sum_{i=0}^{s-1} \lambda_i \quad \text{and} \quad t' = \frac{\sum_{i=0}^{s-1} \lambda_i t_i}{\sum_{i=0}^{s-1} \lambda_i}.$$

Obviously,

$$0 < t' < t. \tag{4}$$

The probability p can be further written as

$$p = \frac{\lambda}{n-t}\left(1 - \frac{\lambda' s}{n-t'}\right) + o(1/n). \tag{5}$$

We now show the lemma by contradiction. Assume that the lemma does not hold, namely $p \geq p_0$, where $p_0 = e/n$ by Lemma 2.2. Let $\delta = (e - \lambda)/e$. Since $1 < \lambda < e$,

we know that $0 < \delta < 1 - 1/e$. Then by ignoring the $o(1/n)$ term and some algebra, it can be shown from (5) that when $t < \delta n$, $p \geq p_0$ is equivalent to

$$t' > n + \frac{\lambda\lambda' s n}{(e - \lambda)n - et} > n.$$

Since $n \geq t$, we have $t' > t$, which is in contradiction with (4). □

In a search tree with nodes generated according to equation (1), only one node on the first depth of the tree has no included arcs. When a subtour with the minimum number of arcs is chosen, the number of children generated is $O(n/\ln(n))$ on average by Lemma 2.3. Thus, the nodes at the first depth have $t = O(n/\ln(n))$ included arcs. Asymptotically $t < \delta n$, where $0 < \delta < 1 - 1/e$ is a constant. Thus, all nodes except one without included arcs on the first depth asymptotically satisfy $t < \delta n$. Similarly, all nodes except one on the i-th depth satisfy $t < \delta n$ when i is no bigger than $O(\ln(n))$. Now consider a node that has no included arcs. Notice that none of the ancestors of the node have included arcs either. By Lemma 3.1, the probability of the AP solution of the node or one of its ancestors being a tour can be asymptotically approximated as e/n by ignoring the $o(1/n)$ term. Thus, the probability p that a node with no included arcs exists and is a leaf node on the d-th depth of the search tree can be approximated by $p = (e/n)(1 - e/n)^d$, which is less than e/n, the probability p_0 that the root node is a leaf. Overall, for nodes at depth $i <$ no bigger than $O(\ln(n))$, the probabilities that they are leaf nodes are asymptotically less than the probability of the root being a leaf.

Recall that the probability that the AP solution of a subproblem selected by BnB subtour elimination will yield an optimal tour is less than the probability that the node in the search tree will become a leaf node. If the depth of the node that generates the optimal tour is greater than $\ln(n)$, or a node with depth greater than $\ln(n)$ is expanded, then $\ln(n)$ nodes must be expanded whose probabilities of generating an optimal tour are less than p_0. Consider the case when the depth of the node generating an optimal tour is less than $\ln(n)$, and no node at depth greater than $\ln(n)$ is generated. Assume further that Bellmore and Malone's argument holds, i.e., only a polynomial number of nodes need to be expanded, which is greater than $\ln(n)$. Then the probabilities that all these expanded nodes generate the optimal tour are asymptotically less than p_0, which contradicts the assumption used in the polynomial argument. Therefore, we have the following result, which disproves Bellmore and Malone's polynomial argument.

Theorem 3.1 *Let $X_0, X_1, X_2, \cdots$ be the subproblems expanded by BnB subtour elimination on a random ATSP, and p_i be the probability that the modified AP solution of X_i is a complete tour, for $i = 0, 1, 2, \cdots$. BnB subtour elimination expands more than $\ln(n)$ number of subproblems X_i with probabilities $p_i < p_0$ for $i \geq 1$.* □

4 Conclusion

For two decades, the question of whether the expected complexity of the ATSP under BnB subtour elimination is polynomial or exponential has remained open. We have shown in this paper that the polynomial argument is not valid.

The polynomial argument critically depends on the assumptions that for a random ATSP, the subproblems considered by BnB subtour elimination are independent of each other, and $p_i \geq p_0$, for $i \geq 1$, where p_i is the probability that the solution of the assignment problem to the i-th subproblem selected by the algorithm is a complete tour. We have proved that, even if the subproblems selected by BnB subtour elimination are assumed to be independent of each other, the algorithm expands more than $\ln(n)$ number of subproblems X_i for which $p_i < p_0$.

Appendix

Lemma 3.1 *Given an $n \times n$ random cost matrix, let $q(n, s, t)$ be the probability that the solution of a modified AP, which has $s < n$ excluded arcs and t included arcs, is a tour. Then $q(n, s, t)$ is asymptotically*

$$e/n - o(1/n) < q(n, s, t) < e/n + o(1/n), \quad \textit{when } t = 0 \tag{6}$$

$$q(n, s, t) = \frac{\lambda}{n - t} + o(1/n), \quad \textit{when } t > 0 \tag{7}$$

where λ, $1 < \lambda < e$, is a constant.

Proof: Denote by E and I the excluded and included sets. Following the same argument used in the proof of Lemma 2.3, it can be easily shown that solving the AP with constraints E and I is equivalent to arbitrarily selecting a feasible permutation from all feasible ones among which there are some cyclic permutations. In other words, the probability $q(n, s, t)$ of the AP solution being a cyclic permutation is equal to the ratio of the total number of cyclic permutations to the total number of feasible permutations under the constraints. Let $R(n, s, t)$ be the number of cyclic permutations and $Q(n, s, t)$ the number of feasible permutations with constraints E and I, then

$$q(n, s, t) = \frac{R(n, s, t)}{Q(n, s, t)}. \tag{8}$$

(a) First consider $q(n, 0, t)$, the case when there exist some included arcs, but no excluded arcs. Assume initially that no two arcs of I share a common vertex. Consequently, the number of included arcs must be less than half of n, the number of cities, i.e., $t \leq \lfloor n/2 \rfloor$. For an included arc (i, j), since i must be assigned to j,

all out-going arcs from i and in-coming arcs to j except (i,j) can be ignored. Thus the AP with (i,j) included can be solved by treating cities i and j as one vertex. If all arcs in I are thus considered as cities, by Lemma 2.1 there are $(n-t)!/e$ feasible permutations that have no self loops, which also exclude the feasible assignments of j to i for included arcs $(i,j) \in I$. We count the number of permutations with this type of legal 'self loop', namely i to j and j to i for all $(i,j) \in I$, as follows. For $(i,j) \in I$, assign j to i, and considering the other $t-1$ included arcs as cities, there are $(n-2-(t-1))!/e$ permutations, and the total number of choices for (i,j) is $\binom{t}{1}$. Similarly, for two arcs (i,j) and (i',j'), assign j and j' to i and i' respectively. There are $(n-4-(t-2))!/e$ feasible solutions, and the number of combinations of these two arcs is $\binom{t}{2}$. Thus, the total number of feasible permutations with included set I is

$$Q(n,0,t) = \sum_{k=0}^{t} \binom{t}{k} \frac{(n-t-k)!}{e}. \tag{9}$$

Obviously,

$$Q(n,0,t) \geq (n-t)!/e. \tag{10}$$

When $t \leq \lfloor n/2 \rfloor$, it can be simply shown that

$$\binom{t}{k} (n-t-k)! \leq \frac{(n-t)!}{k!}. \tag{11}$$

Thus, by substituting (11) into (9) we have

$$Q(n,0,t) \leq \frac{(n-t)!}{e} \sum_{k=0}^{t} \frac{1}{k!} < \frac{(n-t)!}{e} \sum_{k=0}^{\infty} \frac{1}{k!} = (n-t)!. \tag{12}$$

Now consider the case when there are r cities which are shared by two arcs in I. Two arcs sharing a common vertex can be treated as one arc connecting three cities, and the shared vertex can be simply ignored. The problem is then equivalent to one with $n-r$ cities and $t-r$ included arcs. Following the previous reasoning, we have

$$Q(n,0,t) = \sum_{k=0}^{t-r} \binom{t-r}{k} \frac{(n-t-k)!}{e}. \tag{13}$$

Obviously, inequality (10) is still valid in this case, and the proof of (12) follows. When there are t included arcs and r cities commonly shared by the included arcs,

the number of distinct cities supporting the included arcs is $2t - r$ which must not be larger than n, i.e.,

$$2t - r \leq n. \tag{14}$$

With inequality (14), it can be shown that

$$\binom{t-r}{k}(n-t-k)! \leq \frac{(n-t)!}{k!}. \tag{15}$$

Then the upper bound in (12) follows by substituting (15) into (9). Overall, for a given included set I with t arcs, we have

$$(n-t)!/e < Q(n,0,t) < (n-t)!. \tag{16}$$

The number of cyclic permutations is simply

$$R(n,0,t) = (n-t-1)!. \tag{17}$$

Finally, by (8), (16) and (17), we write

$$\frac{1}{n-t} < q(n,0,t) < \frac{e}{n-t}. \tag{18}$$

(b) Now consider $q(n,s,0)$, the case when there are some excluded arcs, and there is no included arc. For this case, we have

$$Q(n,s,0) = n!/e - Q(n,0,s). \tag{19}$$

This is the difference between the number of feasible solutions without any constraints, which is $n!/e$, and the number of feasible solutions if we include the excluded arcs in the solutions. Combining (19) with (16), we obtain

$$n!/e - (n-s)! < Q(n,s,0) < n!/e - (n-s)!/e. \tag{20}$$

To compute $R(n,s,0)$, we first assume that no two arcs in E have a common vertex. $R(n,s,0)$ is the number of cyclic permutations under the constraints of the excluded set E. There are $R(n,0,0)$ cyclic permutations without the constraints of E. In other words, we over-counted the number of feasible cyclic permutations by including those arcs that were excluded before. The number of cyclic permutations with one particular excluded arc included is $R(n,0,1)$, and the number of choices for this arc is $\binom{s}{1}$. Similarly, the number of cyclic permutations with two

particular excluded arcs included is $R(n,0,2)$, and the number of combinations of these two arcs is $\binom{s}{2}$. By the principle of inclusion and exclusion [12], we have

$$R(n,s,0) \;=\; \sum_{k=0}^{s}(-1)^k \binom{s}{k} R(n,0,k).$$

Obviously, $R(n,s,0)$ is bounded by

$$R(n,0,0) - sR(n,0,1) < R(n,s,0) < R(n,0,0). \tag{21}$$

Now we show that (21) still holds when E contains r cities shared by two arcs. The upper bound holds since $R(n,0,0)$ is the number of cyclic permutations without any constraints posed by E. The lower bound is valid because, by the same reason as used in (13), we have

$$R(n,s,0) = \sum_{k=0}^{s-r}(-1)^k \binom{s-r}{k} R(n,0,k)$$

when there are r common cities shared by arcs of E. Furthermore, when $s < n$,

$$\binom{s-r}{k} R(n,0,k) < \binom{s}{k} R(n,0,k) < \binom{s}{1} R(n,0,1).$$

Using (17), (20) and (21), we write

$$\frac{e\left((n-1)! - s(n-2)!\right)}{n! - e(n-s)!} < q(n,s,0) < \frac{e(n-1)!}{n! - (n-s)!}.$$

By some algebra, we obtain the following,

$$\frac{e}{n} - \frac{es}{n(n-1)} < q(n,s,0) < \frac{e}{n} + \frac{es}{n(n(n-1)\cdots(n-s+1)-1)}.$$

When $s < n$, we can write

$$e/n - o(1/n) < q(n,s,0) < e/n + o(1/n), \tag{22}$$

which is (6).

(c) By Lemma 2.1, $q(n,0,0) = e/n$. Thus (22) means that asymptotically, the arcs of E have only secondary influence on the probability that a modified AP solution is a tour when $s < n$. Combining (18) and (22), and the fact that the excluded and included sets are disjoint, we obtain

$$\frac{1}{(n-t)} - o(1/n) < q(n,s,t) < \frac{e}{n-t} + o(1/n). \tag{23}$$

Therefore, there must exist a constant $\lambda \in (1,e)$ such that

$$q(n,s,t) = \frac{\lambda}{n-t} + o(1/n), \tag{24}$$

which is (7). $\square$

Acknowledgment

The author would like to thank Liz Borowsky, Richard Karp, and Richard Korf for helpful discussions related to this work, and comments on an early draft.

References

[1] E. Balas, T.W. McGuire, and P. Toth. Statistical analysis of some traveling salesman problem algorithms. Technical Report Management science research report 501, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA, 1984.

[2] E. Balas and P. Toth. Branch and bound methods. In E.L. Lawler *et. al.*, editor, *The Traveling Salesman Problem*, pages 361–401. John Wiley & Sons, Essex, 1985.

[3] M. Bellmore and J. C. Malone. Pathology of traveling-salesman subtour-elimination algorithms. *Operations Research*, 19:278–307, 1971.

[4] G. Carpaneto and P. Toth. Some new branching and bounding criteria for the asymmetric traveling salesman problem. *Management Science*, 26:736–743, 1980.

[5] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, New York, NY, 1979.

[6] R. M. Karp. A patching algorithm for the nonsymmetric traveling-salesman problem. *SIAM Journal on Computing*, 8:561–573, 1979.

[7] R. M. Karp and J. M. Steele. Probabilistic analysis of heuristics. In E. L. Lawler *et al.*, editor, *The Traveling Salesman Problem*, pages 181–205. John Wiley & Sons, Essex, 1985.

[8] E. L. Lawler, J. K. Lenstra, A. H. G Rinnooy Kan, and D. B. Shmoys. *The Traveling Salesman Problem.* John Wiley & Sons, Essex, 1985.

[9] E. L. Lawler and D. E. Wood. Branch-and-bound methods: A survey. *Operations Research*, 14:699–719, 1966.

[10] J. K. Lenstra and A. H. G. Rinnooy Kan. On the expected performance of branch-and-bound algorithms. *Operations Research*, 26:347–349, 1978.

[11] S. Martello and P. Toth. Linear assignment problems. *Annals of Discrete Mathematics*, 31:259–282, 1987.

[12] J. Riordan. *An Introduction to Combinatorial Analysis.* Princeton University Press, Princeton, N.J., 1978.

[13] D. R. Smith. Random trees and the analysis of branch and bound procedures. *Journal of ACM*, 31:163–188, 1984.

[14] T. H. C. Smith, V. Srinivasan, and G. L. Thompson. Computational performance of three subtour elimination algorithms for solving asymmetric traveling salesman problems. *Annals of Discrete Mathematics*, 1:495–506, 1977.